Các thuật toán dựa theo đại diện

- Với các thuật toán dựa theo đại diện này, chúng ta có một số vấn đề đáng quan tâm.
  - Các tiêu chí khởi động thuật toán.
  - Chọn số nhóm.
  - Ngoại lai.

Các thuật toán gom cụm phân tầng

- Các thuật toán gom cụm phân tầng thường gom cụm dữ liệu với khoảng cách. Tuy nhiên, các hàm khoảng cách thường không bắt buộc phải có.
- Nhiều thuật toán phân tầng sử dụng các phương pháp gom cụm khác dưới dạng subroutine để xây dựng các tầng.
- Một lý do chính để sử dụng các phương pháp phân tầng là các độ mịn gom nhóm khác nhau cho chúng ta thêm các hiểu biết cụ thể theo ứng dụng.

Các thuật toán gom cụm phân tầng

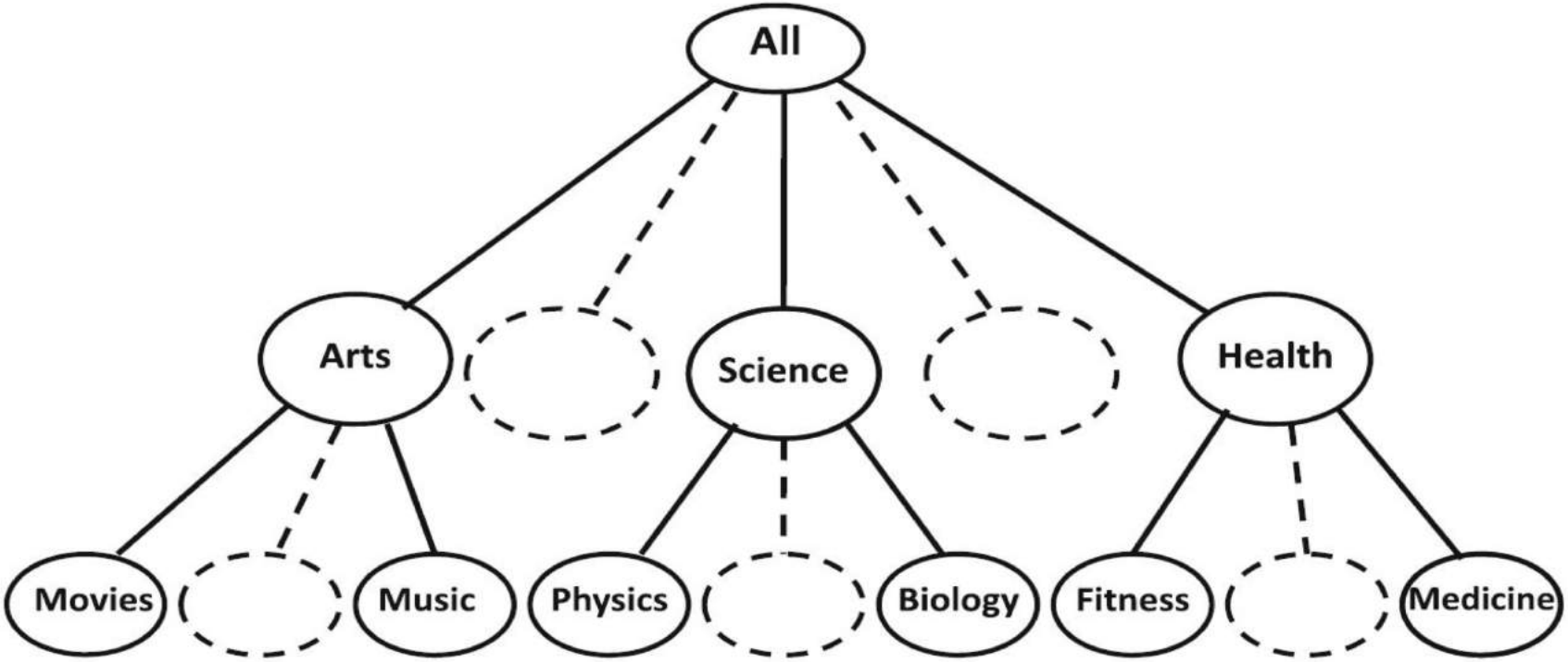


Figure 6.6: Multigranularity insights from hierarchical clustering

Source: Data Mining, Charu C. Aggarwal

Các thuật toán gom cụm phân tầng

- Có 2 loại thuật toán phân tầng chính dựa vào cách cây phân tầng được xây dựng thế nào.
  - Các phương pháp từ dưới lên (agglomerative/kết tụ)
  - Các phương pháp từ trên xuống (divisive)

Phương pháp từ dưới lên (kết tụ)

- Trong các phương pháp này, mỗi điểm dữ liệu bắt đầu với nhóm riêng (nhóm 1 điểm dữ liệu) và được tuần tự kết tụ thành các nhóm bậc cao hơn.

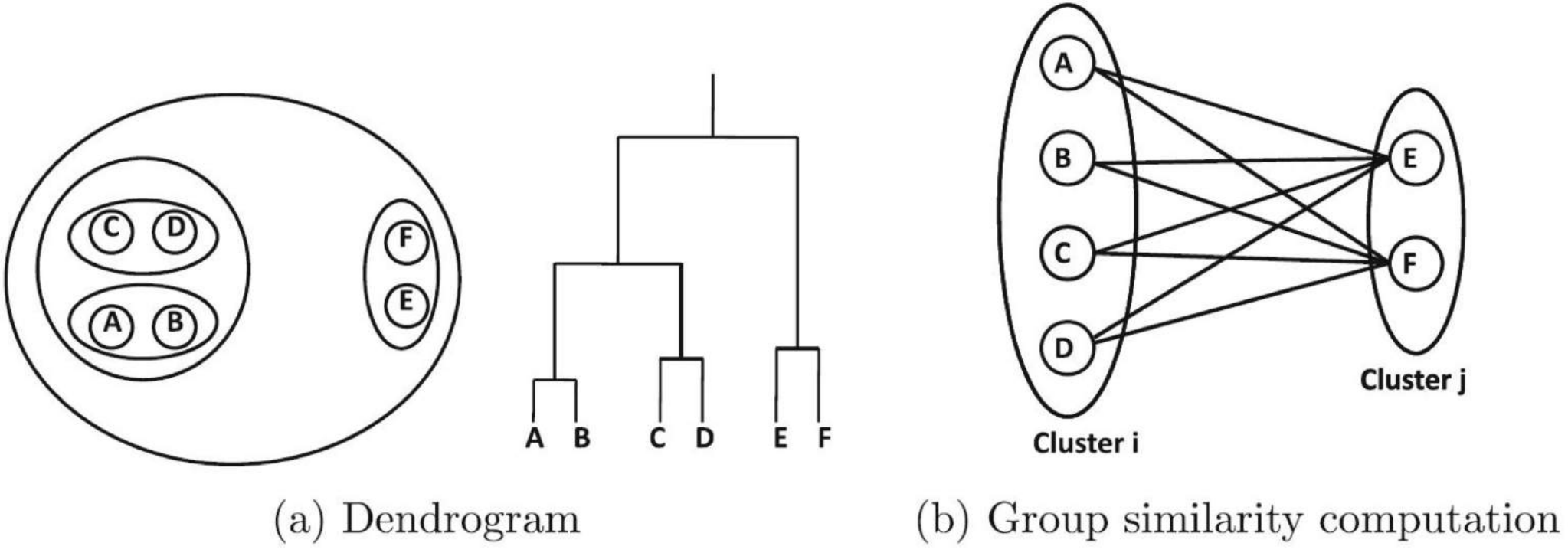
Phương pháp từ dưới lên (kết tụ)

Algorithm *AgglomerativeMerge*(Data: $\mathcal{D}$)
begin
  Initialize $n \times n$ distance matrix M using $\mathcal{D}$;
  repeat
    Pick closest pair of clusters i and j using M;
    Merge clusters i and j;
    Delete rows/columns i and j from M and create
      a new row and column for newly merged cluster;
    Update the entries of new row and column of M;
  until termination criterion;
  return current merged cluster set;
end

Figure 6.7: Generic agglomerative merging algorithm with unspecified merging criterion

Source: Data Mining, Charu C. Aggarwal

Phương pháp từ dưới lên (kết tụ)



(a) Dendrogram (b) Group similarity computation

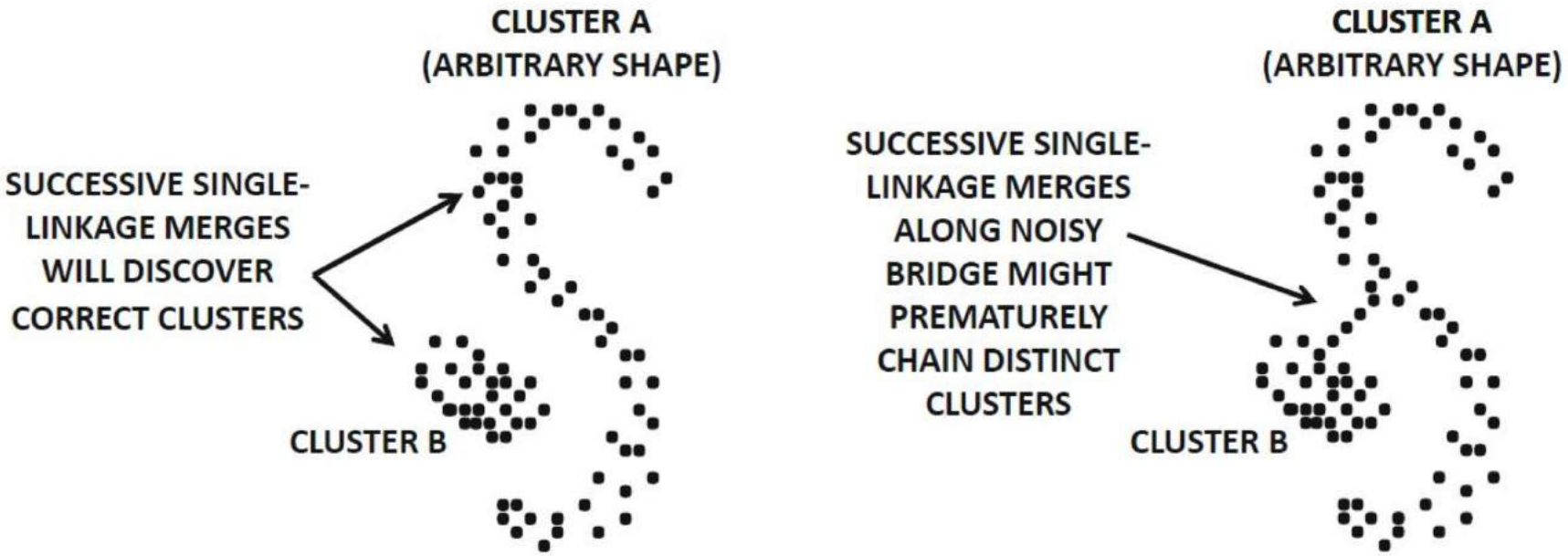
Figure 6.8: Illustration of hierarchical clustering steps

Source: Data Mining, Charu C. Aggarwal

Phương pháp từ dưới lên (kết tụ)

- Trong các phương pháp này, chúng ta có một số tiêu chí như sau cho việc gộp các nhóm trong mỗi bước của thuật toán.
  - Best (single) linkage.
  - Worst (complete) linkage.
  - Group-average linkage.
  - Closest centroid.
  - Các tiêu chí dựa theo phương sai.
  - Phương pháp ward.

Phương pháp từ dưới lên (kết tụ)



(a) Good case with no noise

(b) Bad case with noise

Figure 6.9: Good and bad cases for single-linkage clustering

Source: Data Mining, Charu C. Aggarwal

Phương pháp từ trên xuống (divisive)

- Các phương pháp phân tầng từ trên xuống có thể được xem là các thuật toán meta mà có thể dùng subroutine là hầu hết các thuật toán gom cụm.

- Với cách tiếp cận từ trên xuống, chúng ta có thể có kiểm soát tốt hơn trên cấu trúc cây phân tầng.

Các thuật toán gom cụm phân tầng

Phương pháp từ trên xuống (divisive)

Algorithm *GenericTopDownClustering*(Data: $\mathcal{D}$, Flat Algorithm: $\mathcal{A}$)
begin
  Initialize tree $\mathcal{T}$ to root containing $\mathcal{D}$;
  repeat
    Select a leaf node L in $\mathcal{T}$ based on pre-defined criterion;
    Use algorithm $\mathcal{A}$ to split L into $L_1 \ldots L_k$;
    Add $L_1 \ldots L_k$ as children of L in $\mathcal{T}$;
  until termination criterion;
end

Figure 6.10: Generic top-down meta-algorithm for clustering

Source: Data Mining, Charu C. Aggarwal

Các thuật toán dựa theo mô hình xác suất

- Các thuật toán như chúng ta đã tìm hiểu mà mỗi điểm dữ liệu được gom xác định gán vào một cụm cụ thể gọi là hard clustering algorithm.

- Trong khi đó, các thuật toán dựa theo mô hình xác suất là các ***soft*** *clustering algorithm* mà mỗi điểm dữ liệu được gán xác suất với nhiều (có thể tất cả) nhóm.

- Kết quả của thuật toán soft có thể chuyển về kết quả hard bằng cách gán các điểm dữ liệu vào nhóm với xác suất cao nhất.

Các thuật toán dựa theo mô hình xác suất

The broad principle of a mixture-based *generative* model is to assume that the data was generated from a mixture of k distributions with probability distributions $\mathcal{G}_1 \ldots \mathcal{G}_k$. Each distribution $\mathcal{G}_i$ represents a cluster and is also referred to as a *mixture component*. Each data point $\overline{X_i}$, where $i \in \{1 \ldots n\}$, is generated by this mixture model as follows:

1. Select a mixture component with prior probability $\alpha_i = P(\mathcal{G}_i)$, where $i \in \{1 \ldots k\}$. Assume that the rth one is selected.

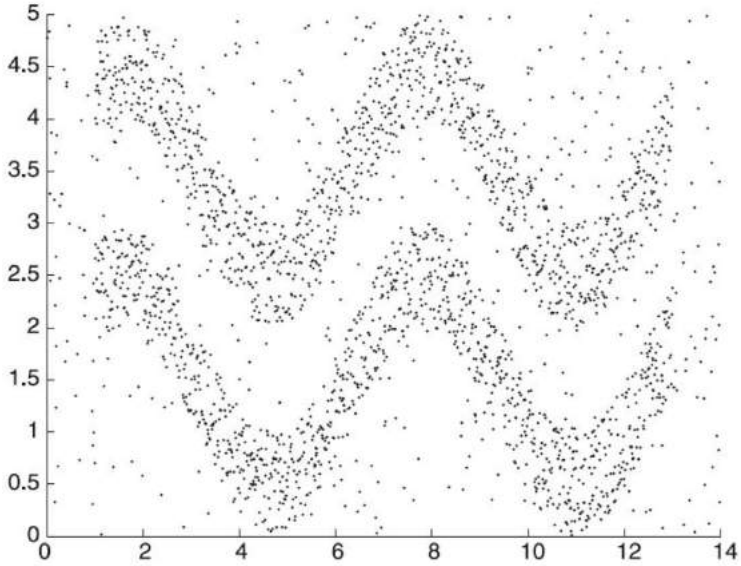
2. Generate a data point from $\mathcal{G}_r$.

Source: Data Mining, Charu C. Aggarwal

Các thuật toán dựa theo lưới và dựa theo mật độ

- Một vấn đề quan trọng với các thuật toán dựa theo khoảng cách hoặc các phương pháp xác suất là hình dáng của các nhóm đã được quy định ngầm với hàm khoảng cách hoặc phân phối xác suất.

- Đặc tính này sẽ không phù hợp với một số ứng dụng cần các nhóm có hình dạng bất kì.



(a) Arbitrarily-shaped clusters

Source: Data Mining, Charu C. Aggarwal

Các thuật toán dựa theo lưới và dựa theo mật độ

- Với các tình huống thế này thì các phương pháp dựa theo mật độ rất hữu ích.
- Ý tưởng chính của phương pháp này là xác định các vùng dày đặc (mật độ cao) trong dữ liệu. Các vùng này sẽ là các “khối xây dựng” cho các cụm với hình dáng bất kì.
- Tùy thuộc vào việc lựa chọn các “khối xây dựng” mà chúng ta có các biến thể.

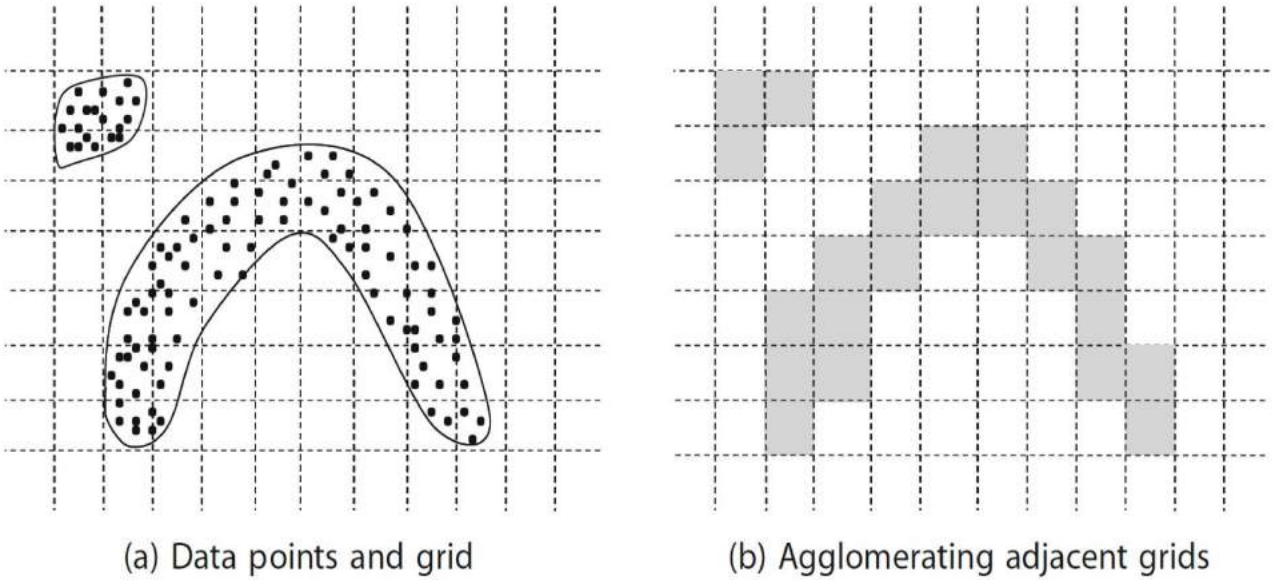
Các thuật toán dựa theo lưới và dựa theo mật độ

- Với các tình huống thế này thì các phương pháp dựa theo mật độ rất hữu ích.
- Ý tưởng chính của phương pháp này là xác định các vùng dày đặc (mật độ cao) trong dữ liệu. Các vùng này sẽ là các “khối xây dựng” cho các cụm với hình dáng bất kì.
- Tùy thuộc vào việc lựa chọn các “khối xây dựng” mà chúng ta có các biến thể, bao gồm các phương pháp dựa theo lưới.

Các thuật toán dựa theo lưới

- Với các phương pháp này, dữ liệu được rời rạc hóa thành một số các khoảng (thường là cùng chiều rộng).
- Các hypercube từ việc rời rạc hóa này chính là các “khối xây dựng” cho thuật toán.
- Ở đây chúng ta có một threshold mật độ được dùng để xác định xem tập con nào của các hypercube này dày đặc (có mật độ cao).

Các thuật toán dựa theo lưới



(a) Data points and grid

(b) Agglomerating adjacent grids

Figure 6.13: Agglomerating adjacent grids

Source: Data Mining, Charu C. Aggarwal

Các thuật toán dựa theo lưới

Algorithm *GenericGrid*(Data: $\mathcal{D}$, Ranges: p, Density: τ)
begin
  Discretize each dimension of data $\mathcal{D}$ into p ranges;
  Determine dense grid cells at density level τ;
  Create graph in which dense grids are connected if they are adjacent;
  Determine connected components of graph;
  return points in each connected component as a cluster;
end

Figure 6.12: Generic grid-based algorithm

Source: Data Mining, Charu C. Aggarwal

Các thuật toán dựa theo mật độ - DBSCAN

The *DBSCAN* approach works on a very similar principle as grid-based methods. However, unlike grid-based methods, the density characteristics of data points are used to merge them into clusters. Therefore, the individual data points *in dense regions* are used as building blocks after classifying them on the basis of their density.

The density of a data point is defined by the number of points that lie within a radius *Eps* of that point (including the point itself). The densities of these spherical regions are used to classify the data points into *core*, *border*, or *noise* points. These notions are defined as follows:

1. *Core point:* A data point is defined as a *core* point, if it contains[4] at least τ data points.

2. *Border point:* A data point is defined as a *border* point, if it contains less than τ points, but it also contains at least one core point within a radius *Eps*.
3. *Noise point:* A data point that is neither a core point nor a border point is defined as a *noise* point.

Các thuật toán dựa theo mật độ - DBSCAN

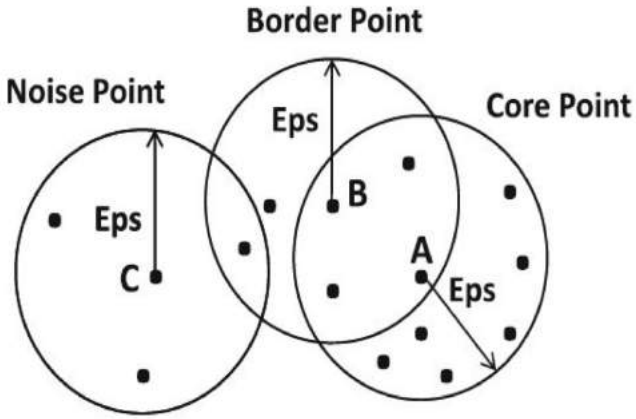


Figure 6.16: Examples of core, border, and noise points

Source: Data Mining, Charu C. Aggarwal

Các thuật toán dựa theo mật độ - DBSCAN

Algorithm *DBSCAN*(Data: $\mathcal{D}$, Radius: *Eps*, Density: τ)
begin
  Determine core, border and noise points of $\mathcal{D}$ at level (Eps, τ);
  Create graph in which core points are connected
    if they are within *Eps* of one another;
  Determine connected components in graph;
  Assign each border point to connected component
    with which it is best connected;
  return points in each connected component as a cluster;
end

Figure 6.15: Basic *DBSCAN* algorithm

Source: Data Mining, Charu C. Aggarwal

Các thuật toán dựa theo mật độ - DENCLUE

The *DENCLUE* algorithm is based on firm statistical foundations that are rooted in kernel-density estimation. Kernel-density estimation can be used to create a smooth profile of the density distribution. In kernel-density estimation, the density $f(\overline{X})$ at coordinate $\overline{X}$ is defined as a sum of the influence (kernel) functions $K(\cdot)$ over the n different data points in the database $\mathcal{D}$:

$$f(\overline{X}) = \frac{1}{n} \sum_{i=1}^{n} K(\overline{X} - \overline{X_i}). \tag{6.18}$$

A wide variety of kernel functions may be used, and a common choice is the Gaussian kernel. For a d-dimensional data set, the Gaussian kernel is defined as follows:

$$K(\overline{X} - \overline{X_i}) = \left(\frac{1}{h\sqrt{2\pi}} \right)^d e^{-\frac{||\overline{X} - \overline{X_i}||^2}{2 \cdot h^2}}. \tag{6.19}$$

Source: Data Mining, Charu C. Aggarwal

Các thuật toán dựa theo mật độ - DENCLUE

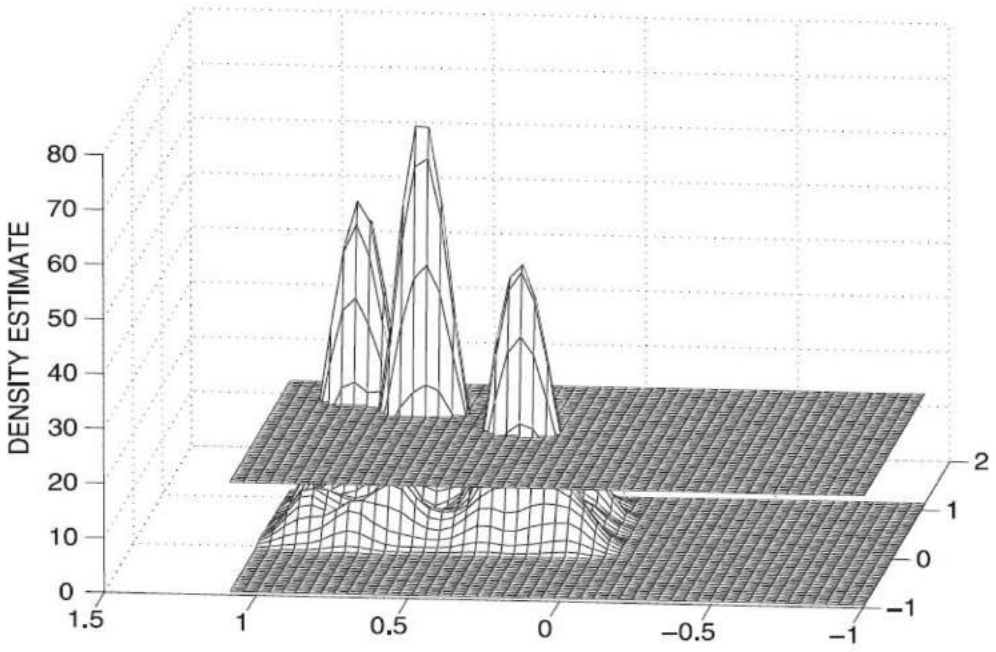


Figure 6.18: Density-based profile with lower density threshold

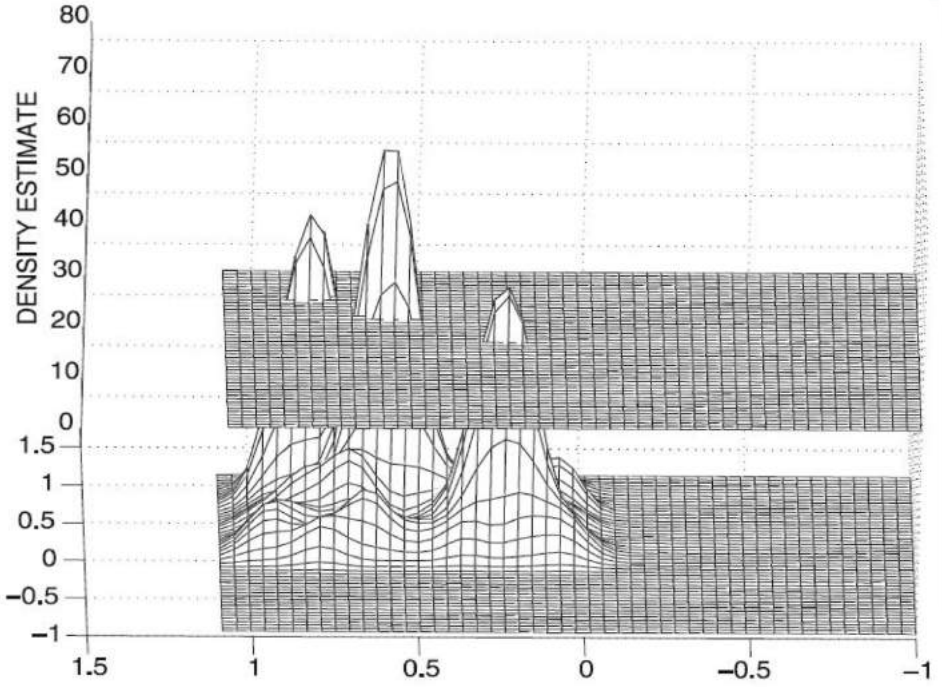


Figure 6.19: Density-based profile with higher density threshold

Source: Data Mining, Charu C. Aggarwal

Phân tích gom nhóm (tiếp theo)

Các thuật toán dựa theo đồ thị

- Các thuật toán dựa theo đồ thị cung cấp một meta-framework chung mà trong đó gần như tất cả kiểu dữ liệu đều có thể được gom nhóm.

- Ý quan trọng cần để ý ở đây là gần như tất cả kiểu dữ liệu đều có thể được biến đổi thành đồ thị tương đồng để thực hiện phân tích.

Các thuật toán dựa theo đồ thị

This transformation will be revisited in the following discussion. The notion of pairwise similarity is defined with the use of a *neighborhood graph*. Consider a set of data objects $\mathcal{O} = \{O_1 \ldots O_n\}$, on which a neighborhood graph can be defined. Note that these objects can be of any type, such as time series or discrete sequences. The main constraint is that it should be possible to define a distance function on these objects. The neighborhood graph is constructed as follows:

1. A single node is defined for each object in $\mathcal{O}$. This is defined by the node set N, containing n nodes, where the node i corresponds to the object O_i.

Source: Data Mining, Charu C. Aggarwal

Các thuật toán dựa theo đồ thị

This transformation will be revisited in the following discussion. The notion of pairwise similarity is defined with the use of a *neighborhood graph*. Consider a set of data objects $\mathcal{O} = \{O_1 \ldots O_n\}$, on which a neighborhood graph can be defined. Note that these objects can be of any type, such as time series or discrete sequences. The main constraint is that it should be possible to define a distance function on these objects. The neighborhood graph is constructed as follows:

2. An edge exists between O_i and O_j, if the distance $d(O_i, O_j)$ is less than a particular threshold ϵ. A better approach is to compute the k-nearest neighbors of both O_i and O_j, and add an edge when either one is a k-nearest neighbor of the other. The weight w_{ij} of the edge (i, j) is equal to a kernelized function of the distance between the objects O_i and O_j, so that larger weights indicate greater similarity. An example is the *heat kernel*, which is defined in terms of a parameter t:

$$w_{ij} = e^{-d(O_i,O_j)^2/t^2}. \tag{6.25}$$

For multidimensional data, the Euclidean distance is typically used to instantiate $d(O_i, O_j)$.

Source: Data Mining, Charu C. Aggarwal

Các thuật toán dựa theo đồ thị

This transformation will be revisited in the following discussion. The notion of pairwise similarity is defined with the use of a *neighborhood graph*. Consider a set of data objects $\mathcal{O} = \{O_1 \ldots O_n\}$, on which a neighborhood graph can be defined. Note that these objects can be of any type, such as time series or discrete sequences. The main constraint is that it should be possible to define a distance function on these objects. The neighborhood graph is constructed as follows:

3. (Optional step) This step can be helpful for reducing the impact of local density variations such as those discussed in Fig. 6.14. Note that the quantity $deg(i) = \sum_{r=1}^{n} w_{ir}$ can be viewed as a proxy for the local kernel-density estimate near object O_i. Each edge weight w_{ij} is normalized by dividing it with $\sqrt{deg(i) \cdot deg(j)}$. Such an approach ensures that the clustering is performed after normalization of the similarity values with local densities. This step is not essential when algorithms such as normalized spectral clustering are used for finally clustering nodes in the neighborhood graph. This is because spectral clustering methods perform a similar normalization under the covers.

Source: Data Mining, Charu C. Aggarwal

Các thuật toán dựa theo đồ thị

- Sau khi có đươc đồ thị từ quá trình xây dựng như trước, chúng ta có thể dùng các thuật toán gom nhóm thích hợp cho các đồ thị này.

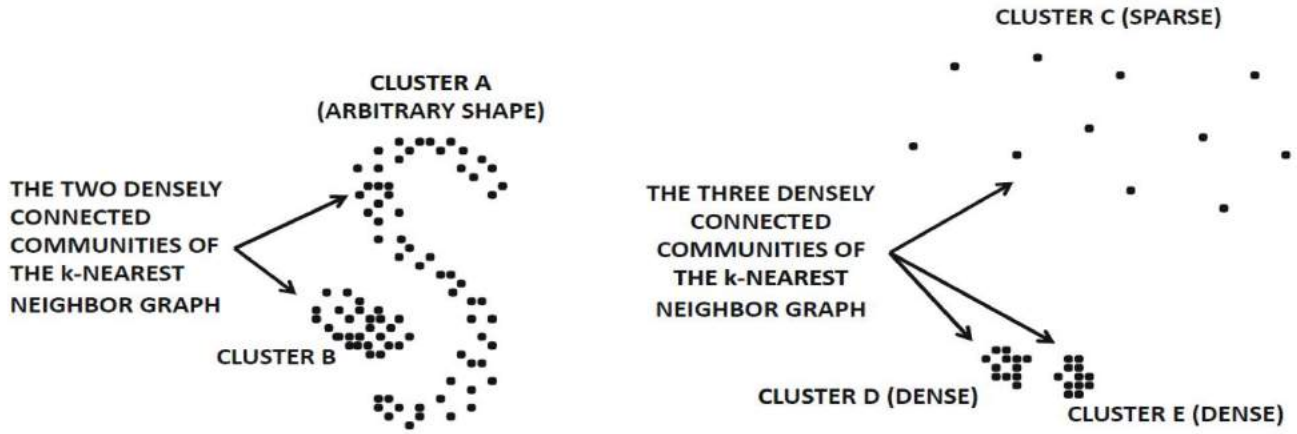
Algorithm *GraphMetaFramework*(Data: $\mathcal{D}$)
begin
  Construct the neighborhood graph G on $\mathcal{D}$;
  Determine clusters (communities) on the nodes in G;
  return clusters corresponding to the node partitions;
end

Figure 6.20: Generic graph-based meta-algorithm

Source: Data Mining, Charu C. Aggarwal

Các thuật toán dựa theo đồ thị

- Một tính chất thú vị của các thuật toán dựa theo đồ thị là các nhóm với hình dạng bất kì có thể được tìm ra do các đồ thị này chứa thông tin về khoảng cách địa phương.



(a) Varying cluster shape (b) Varying cluster density

Figure 6.21: The merits of the k-nearest neighbor graph for handling clusters of varying shape and density

Source: Data Mining, Charu C. Aggarwal

Phân tách ma trận không âm

Nonnegative matrix factorization (*NMF*) is a dimensionality reduction method that is tailored to clustering. In other words, it embeds the data into a latent space that makes it more amenable to clustering. This approach is suitable for data matrices that are *nonnegative* and *sparse*. For example, the $n \times d$ document-term matrix in text applications always contains non-negative entries. Furthermore, because most word frequencies are zero, this matrix is also sparse.

Source: Data Mining, Charu C. Aggarwal

Phân tách ma trận không âm

Nonnegative matrix factorization creates a new *basis system* for data representation, as in all dimensionality reduction methods. However, a distinguishing feature of *NMF* compared to many other dimensionality reduction methods is that the basis system does not necessarily contain orthonormal vectors. Furthermore, the basis system of vectors and the coordinates of the data records in this system are non-negative. The non-negativity of the representation is highly interpretable and well-suited for clustering. Therefore, non-negative matrix factorization is one of the dimensionality reduction methods that serves the dual purpose of enabling data clustering.

Source: Data Mining, Charu C. Aggarwal

Phân tách ma trận không âm

- Với ma trận dữ liệu D kích thước n × d, ở đây chúng ta sẽ tìm 2 ma trận U (n × k) và V (d × k) không âm sao cho hàm mục tiêu sau đạt cực tiểu.

$$J = \frac{1}{2}||D - UV^T||^2.$$

- Bài toán tối ưu này có thể được dùng cho mục đích phân rã sau

$$D \approx UV^T.$$

Source: Data Mining, Charu C. Aggarwal

Phân tách ma trận không âm

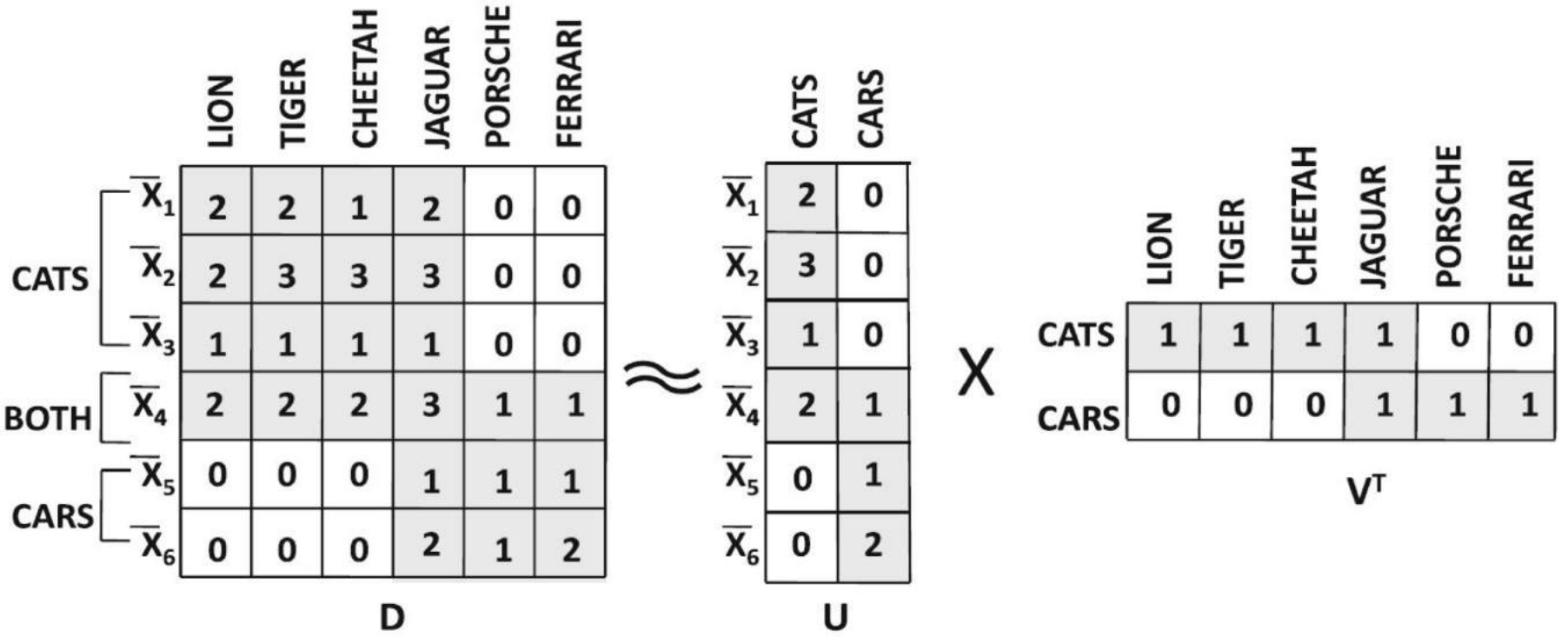


Figure 6.22: An example of non-negative matrix factorization

Source: Data Mining, Charu C. Aggarwal

Phân tách ma trận không âm

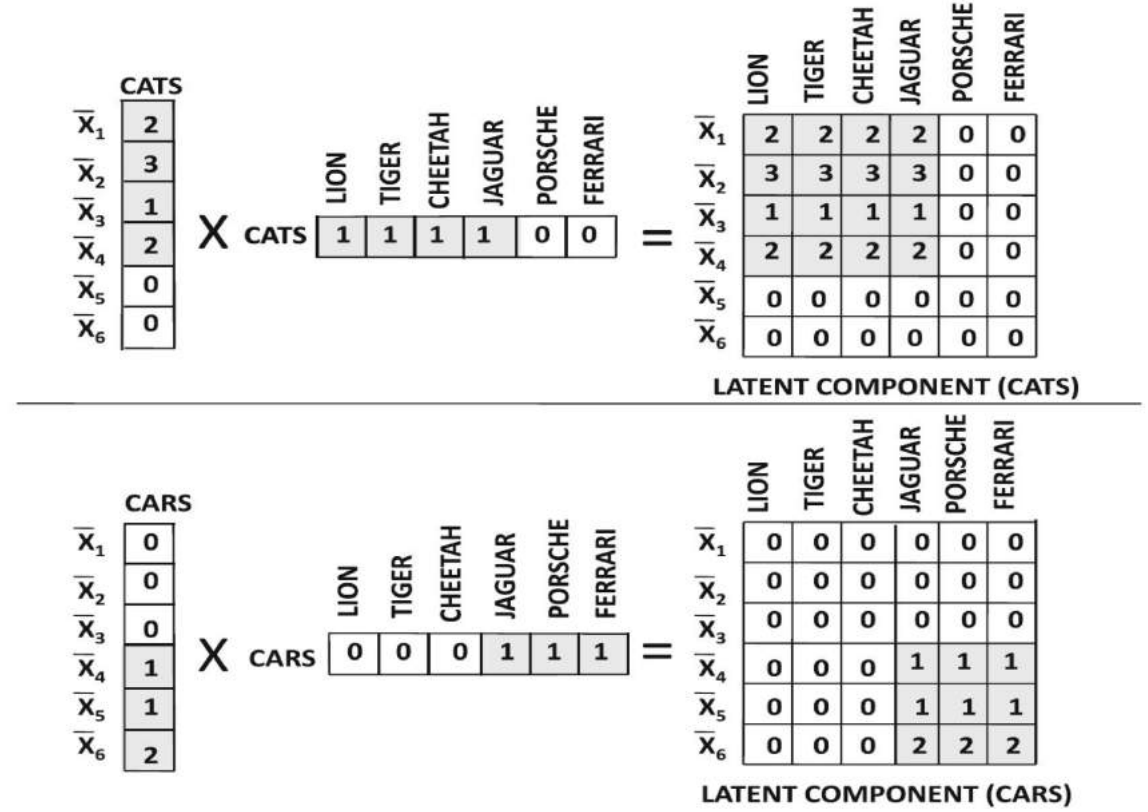


Figure 6.23: The interpretable matrix decomposition of NMF

Source: Data Mining, Charu C. Aggarwal

Phân tách ma trận không âm

One interesting observation about the matrix factorization technique is that it can also be used to determine word-clusters instead of document clusters. Just as the columns of V provide a basis that can be used to discover document clusters, one can use the columns of U to discover a basis that corresponds to word clusters. Thus, this approach provides complementary insights into spaces where the dimensionality is very large.

Source: Data Mining, Charu C. Aggarwal

Phân tách ma trận không âm

- Với phân rã SVD, ta có $D \approx Q_k \Sigma_k P_k^T.$
- So với phân tách ma trận không âm (NMF) U (n × k) và V (d × k)

$$D \approx UV^T.$$

- Chúng ta có ma trận $Q_k\Sigma_k$ (n × k) tương ứng với U (n × k) và P_k (d × k) tương ứng với V (d × k)
- Khác biệt quan trọng nhất ở đây là SVD có ràng buộc về tính trực giao với các vector cơ sở thay vì ràng buộc không âm của NMF

Source: Data Mining, Charu C. Aggarwal

Phân tách ma trận không âm

- Ngoài ra, các biến thể phân tách ma trận khác nhau cũng cho các lợi ích ứng dụng khác nhau.

Phân tách ma trận không âm

1. The latent factors in *NMF* are more easily interpretable for clustering applications, because of non-negativity. For example, in application domains such as text clustering, each of the k columns in U and V can be associated with document clusters and word clusters, respectively. The magnitudes of the non-negative (transformed) coordinates reflect which concepts are strongly expressed in a document. This "additive parts" representation of *NMF* is highly interpretable, especially in domains such as text, in which the features have semantic meaning. This is not possible with *SVD* in which transformed coordinate values and basis vector components may be negative. This is also the reason that *NMF* transformations are more useful than those of *SVD* for clustering. Similarly, the probabilistic forms of non-negative matrix factorization, such as *PLSA*, are also used commonly for clustering. It is instructive to compare the example of Fig. 6.22, with the *SVD* of the same matrix at the end of Sect. 2.4.3.2 in Chap. 2. Note that the *NMF* factorization is more easily interpretable.

Source: Data Mining, Charu C. Aggarwal

Phân tách ma trận không âm

- Ngoài ra, các biến thể phân tách ma trận khác nhau cũng cho các lợi ích ứng dụng khác nhau.

2. Unlike *SVD*, the k latent factors of *NMF* are not orthogonal to one another. This is a disadvantage of *NMF* because orthogonality of the axis-system allows intuitive interpretations of the data transformation as an axis-rotation. It is easy to project *out-of-sample* data points (i.e., data points not included in D) on an orthonormal basis system. Furthermore, distance computations between transformed data points are more meaningful in *SVD*.

Source: Data Mining, Charu C. Aggarwal

Phân tách ma trận không âm

- Ngoài ra, các biến thể phân tách ma trận khác nhau cũng cho các lợi ích ứng dụng khác nhau.

3. The addition of a constraint, such as non-negativity, to any optimization problem usually reduces the quality of the solution found. However, the addition of orthogonality constraints, as in *SVD*, do not affect the *theoretical* global optimum of the *unconstrained* matrix factorization formulation (see Exercise 13). Therefore, *SVD* provides better rank-k approximations than *NMF*. Furthermore, it is much easier *in practice* to determine the global optimum of *SVD*, as compared to unconstrained matrix factorization for matrices that are completely specified. Thus, *SVD* provides one of the alternate global optima of unconstrained matrix factorization, which is computationally easy to determine.

Source: Data Mining, Charu C. Aggarwal

Phân tách ma trận không âm

- Ngoài ra, các biến thể phân tách ma trận khác nhau cũng cho các lợi ích ứng dụng khác nhau.

4. *SVD* is generally hard to implement for incomplete data matrices as compared to many other variations of matrix factorization. This is relevant in recommender systems where rating matrices are incomplete. The use of latent factor models for recommendations is discussed in Sect. 18.5.5 of Chap. 18.

Source: Data Mining, Charu C. Aggarwal

Đánh giá gom cụm

- Với các cụm có được sau quá trình gom cụm, chúng ta cũng quan tâm việc đánh giá chất lượng thông qua việc đánh giá gom cụm.
- Khi chúng ta không có các dữ liệu đánh nhãn do đặc thù không giám sát của bài toán gom cụm, chúng ta cần sử dụng các tiêu chí đánh giá trong.
- Trong trường hợp chúng ta có dữ liệu đánh nhãn, chúng ta có thể sử dụng các tiêu chí đánh giá ngoài.
- Cả 2 hướng này đều có các hạn chế quan trọng đáng lưu ý.

Đánh giá gom cụm

- Chúng ta có một số tiêu chí đánh giá trong như sau.

1. *Sum of square distances to centroids:* In this case, the centroids of the different clusters are determined, and the sum of squared (SSQ) distances are reported as the corresponding objective function. Smaller values of this measure are indicative of better cluster quality. This measure is obviously more optimized to distance-based algorithms, such as k-means, as opposed to a density-based method, such as *DBSCAN*. Another problem with SSQ is that the absolute distances provide no meaningful information to the user about the quality of the underlying clusters.

Source: Data Mining, Charu C. Aggarwal

Đánh giá gom cụm

- Chúng ta có một số tiêu chí đánh giá trong như sau.

2. *Intracluster to intercluster distance ratio:* This measure is more detailed than the SSQ measure. The idea is to sample r pairs of data points from the underlying data. Of these, let P be the set of pairs that belong to the same cluster found by the algorithm. The remaining pairs are denoted by set Q. The average intercluster distance and intracluster distance are defined as follows:

$$Intra = \sum_{(\overline{X_i},\overline{X_j})\in P} dist(\overline{X_i},\overline{X_j})/|P| \tag{6.43}$$

$$Inter = \sum_{(\overline{X_i},\overline{X_j})\in Q} dist(\overline{X_i},\overline{X_j})/|Q|. \tag{6.44}$$

Then the ratio of the average intracluster distance to the intercluster distance is given by $Intra/Inter$. Small values of this measure indicate better clustering behavior.

Source: Data Mining, Charu C. Aggarwal

Đánh giá gom cụm

- Chúng ta có một số tiêu chí đánh giá trong như sau.

3. *Silhouette coefficient:* Let $Davg_i^{in}$ be the average distance of $\overline{X_i}$ to data points *within* the cluster of $\overline{X_i}$. The average distance of data point $\overline{X_i}$ to the points in each cluster (other than its own) is also computed. Let $Dmin_i^{out}$ represent the minimum of these (average) distances, over the other clusters. Then, the silhouette coefficient S_i *specific to the ith object*, is as follows:

$$S_i = \frac{Dmin_i^{out} - Davg_i^{in}}{\max\{Dmin_i^{out}, Davg_i^{in}\}}. \tag{6.45}$$

The overall silhouette coefficient is the average of the data point-specific coefficients. The silhouette coefficient will be drawn from the range $(-1, 1)$. Large positive values indicate highly separated clustering, and negative values are indicative of some level of "mixing" of data points from different clusters. This is because $Dmin_i^{out}$ will be less than $Davg_i^{in}$ only in cases where data point $\overline{X_i}$ is closer to at least one other cluster than its own cluster. One advantage of this coefficient is that the absolute values provide a good intuitive feel of the quality of the clustering.

Source: Data Mining, Charu C. Aggarwal

Đánh giá gom cụm

- Chúng ta có một số tiêu chí đánh giá trong như sau.

4. *Probabilistic measure:* In this case, the goal is to use a mixture model to estimate the quality of a particular clustering. The centroid of each mixture component is assumed to be the centroid of each discovered cluster, and the other parameters of each component (such as the covariance matrix) are computed from the discovered clustering using a method similar to the M-step of EM algorithms. The overall log-likelihood of the measure is reported. Such a measure is useful when it is known from domain-specific knowledge that the clusters *ought* to have a specific shape, as is suggested by the distribution of each component in the mixture.

Source: Data Mining, Charu C. Aggarwal

Đánh giá gom cụm

- Chúng ta có một số tiêu chí đánh giá trong như sau.

4. *Probabilistic measure:* In this case, the goal is to use a mixture model to estimate the quality of a particular clustering. The centroid of each mixture component is assumed to be the centroid of each discovered cluster, and the other parameters of each component (such as the covariance matrix) are computed from the discovered clustering using a method similar to the M-step of EM algorithms. The overall log-likelihood of the measure is reported. Such a measure is useful when it is known from domain-specific knowledge that the clusters *ought* to have a specific shape, as is suggested by the distribution of each component in the mixture.

Source: Data Mining, Charu C. Aggarwal

Đánh giá gom cụm

- Với trường hợp chúng ta có dữ liệu đánh nhãn để sử dụng các tiêu chí đánh giá ngoài (VD: trong trường hợp sử dụng synthetic dataset)
- Một cách tiếp cận ở đây là sử dụng ý tưởng về confusion matrix.

Cluster Indices	1	2	3	4
1	97	0	2	1
2	5	191	1	3
3	4	3	87	6
4	0	0	5	195

Figure 6.25: Confusion matrix for a clustering of good quality

Cluster Indices	1	2	3	4
1	33	30	17	20
2	51	101	24	24
3	24	23	31	22
4	46	40	44	70

Figure 6.26: Confusion matrix for a clustering of poor quality

Source: Data Mining, Charu C. Aggarwal